CLAIMS

I claim:

1. In a local server that receives data from one or more remote entities over

a data transport protocol, a method of applying a cryptographically secure hash to

packets from unverified remote entities for preventing denial of service attacks on

lookup tables used to store state information for one or more remote entities, while

maintaining the performance of the local server for packets from verified remote

entities, the method comprising the acts of:

receiving a packet of data from a remote entity that includes connection

identifier information;

hashing at least a portion of the connection identifier information using a first

hash function for determining if state information exists for the remote entity in a table

of verified remote entities;

if the state information for the remote entity does not exist in the table of

verified remote entities, hashing at least a portion of the connection identifier

information using a second hash function that is cryptographically secure for

determining if state information exists for the remote entity in a table of unverified

remote entities:

if the state information for the remote entity exists in the table of unverified

remote entities, comparing secret information provided within the packet of data with

information previously supplied to the remote entity for determining if the remote entity

can be verified such that state information can be moved to the table of verified remote

entities:

- Page 25 -

if state information for the remote entity does not exist in the table of unverified

remote entities; checking whether the local server is a listener that may accept the

packet of data from the remote entity for determining if state information for the remote

entity should be created in the table of unverified remote entities.

2. The method of claim 1, wherein if the state information for the remote

entity does exist in the table of verified remote entities, standard data transport protocol

processing is performed.

3. The method of claim 2, wherein the standard data transport protocol is

transmission control protocol.

4. The method of claim 1, wherein if the state information for the remote

entity exists in the table of unverified remote entities, but the remote entity cannot be

verified, the method further comprises the act of:

checking if the packet includes a synchronization message for determining how

to respond to the unverified remote entity.

5. The method of the claim 4, wherein if the packet of data includes a

synchronization message, the local server responds by either sending a synchronization-

acknowledgement packet or by deleting the packet.

6. The method of the claim 4, wherein if the packet of data does not include

a synchronization message, the local server responds by one or more of the following

deleting the packet, retransmitting the original message to the remote entity or removing

the state information from the table of unverified remote entities.

7. The method of claim 1, wherein the first hash function is also a

cryptographically secured hash function.

8. The method of claim 7, wherein the first and second hash functions are

one of hardware based or software based.

9. The method of claim 1, wherein if state information for the remote entity

does not exist in either the table of verified remote entities or the table of unverified

remote entities, and wherein the server is a listener that may accept the package of data

from the remote entity, the method further comprising the acts of:

creating state information for the remote entity within the table of unverified

remote entities; and

sending a synchronization-acknowledgement packet that includes an initial

sequence number to the remote entity.

10. The method of claim 1, wherein if state information for the remote entity

does not exist in either the table of verified entities or the table of unverified entities,

and the server is not a listener that may accept the package of data from the remote

entity, the method further comprises the act of:

sending a reset command to the remote entity for indicating that the packet was

not verifiable and needs to be resent.

11. The method of claim 1, wherein the remote entity becomes verified by sharing a secret sent to the remote entity by the local server.

12. In a local server that receives data from one or more remote entities over

a data transport protocol, a method of applying a cryptographically secure hash to

packets from unverified remote entities for preventing denial of service attacks on

lookup tables used to store state information for one or more remote entities, while

maintaining the performance of the local server for packets from verified remote

entities, the method comprising:

an act of receiving a packet of data from a remote entity that includes connection

identifier information;

a step for determining if state information exists for the remote entity in a table

of verified remote entities;

if the state information for the remote entity does not exist in the table of

verified remote entities, a step for determining if state information exists for the remote

entity in a table of unverified remote entities;

if the state information exists in the table of unverified remote entities, a step for

determining if the remote entity can be verified such that state information can be

moved to the table of verified remote entities;

if state information does not exist in the table of unverified remote entities; a

step for determining if state information for the remote entity should be created in the

table of unverified remote entities.

13. The method of claim 12, wherein if the state information for the remote

entity does exist in the table of verified remote entities, standard data transport protocol

processing is performed.

14. The method of claim 13, wherein the standard data transport protocol is

transmission control protocol.

15. The method of claim 12, wherein if the state information exists in the

table of unverified remote entities, but the remote entity cannot be verified, the method

further comprises the act of:

checking if the packet includes a synchronization message for determining how

to respond to the unverified remote entity.

16. The method of the claim 15, wherein if the packet of data includes a

synchronization message, the local server responds by either sending a synchronization-

acknowledgement packet or by deleting the packet.

17. The method of the claim 15, wherein if the packet of data does not

include a synchronization message, the local server responds by one or more of the

following deleting the packet, retransmitting the original message to the remote entity

or removing the state information from the table of unverified remote entities.

18. The method of claim 12, wherein the step for determining if state

information exists for the remote entity in the table of verified remote entities includes

the act of hashing at least a portion of the connection identifier information using a first

hash function, and wherein the step for determining if state information exists for the

remote entity in a table of unverified remote entities includes the act of hashing at least

- Page 30 -

a portion of the connection identifier information using a second hash function that is

cryptographically secure.

19. The method of claim 18, wherein the first hash function is also a

cryptographically secured hash function, and wherein the first and second hash

functions are one of either hardware based or software based.

20. The method of claim 12, wherein if state information does not exist in

either the table of verified remote entities or the table of unverified remote entities, and

wherein the step for determining if state information for the remote entity should be

created in the table of unverified remote entities includes the act of checking whether

the local server is a listener that may accept the package of data from the remote entity,

the method further comprising the acts of:

creating state information for the remote entity within the table of unverified

remote entities; and

sending a synchronization-acknowledgement packet that includes an initial

sequence number to the remote entity.

21. The method of claim 12, wherein if state information does not exist in

either the table of verified entities or the table of unverified entities, the step for

determining if state information for the remote entity should be created in the table of

unverified remote entities includes the act of checking whether the local server is a

listener that may accept the package of data from the remote entity, and the server is not

a listener that may accept the package of data from the remote entity, the method further comprising the act of:

sending a reset command to the remote entity for indicating that the packet was not verifiable and needs to be resent.

22. For a local server that receives data from one or more remote entities

over a data transport protocol, a computer program product comprising computer

readable media carrying computer executable instructions that implement a method of

applying a cryptographically secure hash to packets from unverified remote entities for

preventing denial of service attacks on lookup tables used to store state information for

one or more remote entities, while maintaining the performance of the local server for

packets from verified remote entities, the method comprising the acts of:

receiving a packet of data from a remote entity that includes connection

identifier information;

hashing at least a portion of the connection identifier information using a first

hash function for determining if state information exists for the remote entity in a table

of verified remote entities;

if the state information for the remote entity does not exist in the table of

verified remote entities, hashing at least a portion of the connection identifier

information using a second hash function that is cryptographically secure for

determining if state information exists for the remote entity in a table of unverified

remote entities;

if the state information for the remote entity exists in the table of unverified

remote entities, comparing secret information provided within the packet of data with

information previously supplied to the remote entity for determining if the remote entity

can be verified such that state information can be moved to the table of verified remote

entities;

if state information for the remote entity does not exist in the table of unverified

remote entities; checking whether the local server is a listener that may accept the

packet of data from the remote entity for determining if state information for the remote

entity should be created in the table of unverified remote entities.

23. The computer program product of claim 22, wherein if the state

information for the remote entity exists in the table of unverified remote entities, but the

remote entity cannot be verified, the method further comprises the act of:

checking if the packet includes a synchronization message for determining how

to respond to the unverified remote entity.

24. The computer program product of the claim 23, wherein if the packet of

data includes a synchronization message, the local server responds by either sending a

synchronization-acknowledgement packet or by deleting the packet.

25. The computer program product of the claim 23, wherein if the packet of

data does not include a synchronization message, the local server responds by one or

more of the following deleting the packet, retransmitting the original message to the

remote entity or removing the state information from the table of unverified remote

entities.

26. The computer program product of claim 22, wherein the first hash

function is also a cryptographically secured hash function, and wherein the first and

second hash functions are one of hardware based or software based.

- Page 34 -

27. The computer program product of claim 22, wherein if state information

for the remote entity does not exist in either the table of verified remote entities or the

table of unverified remote entities, and wherein the server is a listener that may accept

the package of data from the remote entity, the method further comprising the acts of:

creating state information for the remote entity within the table of unverified

remote entities; and

sending a synchronization-acknowledgement packet that includes an initial

sequence number to the remote entity.

28. The computer program product of claim 22, wherein if state information

for the remote entity does not exist in either the table of verified remote entities or the

table of unverified remote entities, and the server is not a listener that may accept the

package of data from the remote entity, the method further comprises the act of:

sending a reset command to the remote entity for indicating that the packet was

not verifiable and needs to be resent.

29. For a local server that receives data from one or more remote entities

over a data transport protocol, a computer program product comprising computer

readable media carrying computer executable instructions that implement a method of

applying a cryptographically secure hash to packets from unverified remote entities for

preventing denial of service attacks on lookup tables used to store state information for

one or more remote entities, while maintaining the performance of the local server for

packets from verified remote entities, the method comprising:

an act of receiving a packet of data from a remote entity that includes connection

identifier information;

a step for determining if state information exists for the remote entity in a table

of verified remote entities;

if the state information for the remote entity does not exist in the table of

verified remote entities, a step for determining if state information exists for the remote

entity in a table of unverified remote entities;

if the state information exists in the table of unverified remote entities, a step for

determining if the remote entity can be verified such that state information can be

moved to the table of verified remote entities;

if state information does not exist in the table of unverified remote entities; a

step for determining if state information for the remote entity should be created in the

table of unverified remote entities.

30. The computer program product of claim 29, wherein if the state

information exists in the table of unverified remote entities, but the remote entity cannot

be verified, the method further comprises the act of:

checking if the packet includes a synchronization message for determining how to respond to the unverified remote entity.

31. The computer program product of the claim 30, wherein if the packet of

data includes a synchronization message, the local server responds by either sending a

synchronization-acknowledgement packet or by deleting the packet.

32. The computer program product of the claim 30, wherein if the packet of

data does not include a synchronization message, the local server responds by one or

more of the following deleting the packet, retransmitting the original message to the

remote entity or removing the state information from the table of unverified remote

entities.

33. The computer program product of claim 29, wherein the step for

determining if state information exists for the remote entity in the table of verified

remote entities includes the act of hashing at least a portion of the connection identifier

information using a first hash function, and wherein the step for determining if state

information exists for the remote entity in a table of unverified remote entities includes

the act of hashing at least a portion of the connection identifier information using a

second hash function that is cryptographically secure.

34. The computer program product of claim 33, wherein the first hash

function is also a cryptographically secured hash function, and wherein the first and

second hash functions are one of either hardware based or software based.

35. The computer program product of claim 29, wherein if state information

does not exist in either the table of verified remote entities or the table of unverified

remote entities, and wherein the step for determining if state information for the remote

entity should be created in the table of unverified remote entities includes the act of

checking whether the local server is a listener that may accept the package of data from

the remote entity, the method further comprising the acts of:

creating state information for the remote entity within the table of unverified

remote entities; and

sending a synchronization-acknowledgement packet that includes an initial

sequence number to the remote entity.

36. The computer program product of claim 29, wherein if state information

does not exist in either the table of verified remote entities or the table of unverified

remote entities, the step for determining if state information for the remote entity should

be created in the table of unverified remote entities includes the act of checking whether

the local server is a listener that may accept the package of data from the remote entity,

and wherein the server is not a listener that may accept the package of data from the

remote entity, the method further comprises the act of:

sending a reset command to the remote entity for indicating that the packet was

not verifiable and needs to be resent.

37. The computer program product of claim 29, wherein the remote entity

becomes verified by sharing a secret sent to the remote entity by the local server.